

FIGURE 1

```

<?xml version="1.0" encoding="UTF-8"?>
<schema>
  <simpleType name="length">
    <restriction base="string">
      <pattern value="\d{2}:\d{2}:\d{2}"/>
    </restriction>
  </simpleType>

  <simpleType name="str_owner">
    <restriction base="string">
      <length value="40"/>
    </restriction>
  </simpleType>

  <element name="ATTRIBUTES" mediaPlayerEnabled="Y">
    <complexType>
      <sequence>

        <element name="HEADER">
          <complexType>
            <sequence>
              <element name="STATUS" type="int" minOccurs='0'
                maxOccurs='1'/>
              <element name="FILESIZE" type="long" minOccurs='0'
                maxOccurs='1'/>
              <element name="DESCRIPTION" type="string" minOccurs='0'
                maxOccurs='1'>searchable</element>
              <element name="AUTHOR" type="string" minOccurs='0'
                maxOccurs='1'>searchable</element>
              <element name="OWNER" type="str_owner" minOccurs='0'
                maxOccurs='1'>searchable</element>
              <element name="COPYRIGHT" type="string" minOccurs='0'
                maxOccurs='1'/>
              <element name="CREATEDT" type="string" minOccurs='0'
                maxOccurs='1'>searchable</element>
              <element name="THUMBNAIL" type="string" minOccurs='0'
                maxOccurs='1'/>
            </sequence>
          </complexType>
        </element>

        <element name="DETAILS">
          <complexType>
            <sequence>
              <element name="LENGTH" type="length" minOccurs='0'
                maxOccurs='1'/>
              <element name="ENCODERATE" type="string" minOccurs='0'
                maxOccurs='1'/>
              <element name="VIDEOSIZE" type="string" minOccurs='0'
                maxOccurs='1'/>
              <element name="RATING" type="string" minOccurs='0'
                maxOccurs='1'/>
            </sequence>
          </complexType>
        </element>
      </sequence>
    </complexType>
  </element>

```

```

    <element name="HISTORY">
    <complexType>
    <sequence>
        <element name="WORKFLOW" minOccurs='0' maxOccurs='1'>
        <complexType>
        <sequence>
            <element name="WORK" minOccurs='0'>
            <complexType>
            <attribute name="user"
                                type="string"/>

            <attribute name="time"
                                type="string"/>
            </complexType>
        </element>
        </sequence>
        </complexType>
    </element>
    <element name="NOTES" minOccurs='0' maxOccurs='1'>
    <complexType>
    <sequence>
        <element name="NOTE" type="string"
                                minOccurs='0' />

    </sequence>
    </complexType>
    </element>
    </sequence>
    </complexType>
    </element>
</schema>

```

**FIGURE 2**

```

<?xml version="1.0" encoding="utf-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
version="1.0">
  <xsl:output method="html"/>
  <xsl:template match="schema">
    <xsl:apply-templates select="element"/>
  </xsl:template>
  <xsl:template match="element">
    <table border="0" align="left">
      <xsl:apply-templates
        select="complexType/sequence/element"/>
    </table>
  </xsl:template>
  <xsl:template match="complexType/sequence/element">
    <xsl:variable select="@name" name="group"/>
    <tr>
      <td valign="middle" colspan="2">
        <h3>
          <i>
            <xsl:value-of select="$group"/>:</i>
          </h3>
        </td>
      </tr>
    <xsl:for-each select="complexType/sequence/element">
      <xsl:variable select="@name" name="subgroup"/>
      <xsl:if test="(&apos;FILENAME&apos;!=translate(@name,
        &apos;filename&apos;, &apos;FILENAME&apos;)) and
        &apos;FILETYPE&apos;!=translate(@name,
        &apos;filetype&apos;, &apos;FILETYPE&apos;)) and
        &apos;STATUS&apos;!=translate(@name,
        &apos;status&apos;, &apos;STATUS&apos;)) and
        (&apos;string&apos;=translate(@type,
        &apos;STRING&apos;, &apos;string&apos;)) or
        &apos;long&apos;=translate(@type, &apos;LONG&apos;,
        &apos;long&apos;)) or &apos;int&apos;=translate(@type,
        &apos;INT&apos;, &apos;int&apos;)) or
        &apos;length&apos;=translate(@type,
        &apos;LENGTH&apos;, &apos;length&apos;)) or
        &apos;str_owner&apos;=translate(@type,
        &apos;STR_OWNER&apos;, &apos;str_owner&apos;))">
        <tr>
          <td>
            <b>
              <xsl:value-of select="@name"/>
            </b>
          </td>
          <td>
            <xsl:choose>
              <xsl:when
                test="&apos;THUMBNAIL&apos;=transla
                  te(@name, &apos;thumbnail&apos;,
                  &apos;THUMBNAIL&apos;)">
                <xsl:element name="INPUT">
                  <xsl:attribute name="size">37</xsl:attribute>
                  <xsl:attribute
                    name="type">file</xsl:attribute>

```

```

        <xsl:attribute name="name">
            <xsl:value-of
                select="$group"/>/<xsl:value-of
                select="@name"/>
        </xsl:attribute>
        <xsl:attribute name="accept">image/gif,
            image/jpeg</xsl:attribute>
    </xsl:element>
</xsl:when>
<xsl:otherwise>
    <xsl:element name="INPUT">
        <xsl:attribute
            name="size">50</xsl:attribute>
        <xsl:attribute
            name="type">text</xsl:attribute>
        <xsl:attribute name="name">
            <xsl:value-of
                select="$group"/>/<xsl:value-
                of select="@name"/>
        </xsl:attribute>
        <xsl:attribute name="value"/>
    </xsl:element>
</xsl:otherwise>
</xsl:choose>
</td>
</tr>
</xsl:if>

<xsl:for-each select="complexType/sequence/element">
    <xsl:if test="(&apos;FILENAME&apos;!=translate(@name,
        &apos;filename&apos;, &apos;FILENAME&apos;)) and
        &apos;FILETYPE&apos;!=translate(@name,
        &apos;filetype&apos;, &apos;FILETYPE&apos;)) and
        &apos;STATUS&apos;!=translate(@name,
        &apos;status&apos;, &apos;STATUS&apos;)) and
        (&apos;string&apos;=translate(@type,
        &apos;STRING&apos;, &apos;string&apos;)) or
        &apos;long&apos;=translate(@type, &apos;LONG&apos;,
        &apos;long&apos;)) or &apos;int&apos;=translate(@type,
        &apos;INT&apos;, &apos;int&apos;))">
        <tr>
            <td>
                <b>
                    <xsl:value-of select="@name"/>
                </b>
            </td>
            <td>
                <xsl:element name="INPUT">
                    <xsl:attribute name="size">50</xsl:attribute>
                    <xsl:attribute name="type">text</xsl:attribute>
                    <xsl:attribute name="name">
                        <xsl:value-of select="$group"/>/<xsl:value-
                            of select="$subgroup"/>/<xsl:value-of
                            select="@name"/>
                    </xsl:attribute>
                    <xsl:attribute name="value"/>
                </xsl:element>
            </td>
        </tr>
    </xsl:if>
</xsl:for-each>

```

```
        </td>
      </tr>
    </xsl:if>
  </xsl:for-each>

</xsl:for-each>

</xsl:template>
</xsl:stylesheet>
```

**FIGURE 3**

```

<?xml version="1.0" encoding="UTF-8"?>

<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:output method="html"/>

<xsl:template match="ATTRIBUTES">
  <center><h3>File Name: <xsl:value-of
select="HEADER/FILENAME"/><br></br>
      File Type: <xsl:value-of
select="HEADER/FILETYPE"/></h3></center>
  <table width="100%" border="0">
    <xsl:apply-templates select="HEADER"/>
    <xsl:apply-templates select="DETAILS"/>
    <xsl:apply-templates select="HISTORY"/>
  </table>
</xsl:template>

<xsl:template match="HEADER">
  <tr><th colspan="3" align="left">&#160;<i><xsl:value-of
select="name()" /></i></th></tr>
  <tr><xsl:apply-templates select="FILESIZE"/></tr>
  <tr><xsl:apply-templates select="DESCRIPTION"/></tr>
  <tr><xsl:apply-templates select="AUTHOR"/></tr>
  <tr><xsl:apply-templates select="OWNER"/></tr>
  <tr><xsl:apply-templates select="COPYRIGHT"/></tr>
  <tr><xsl:apply-templates select="CREATEDT"/></tr>
</xsl:template>

<xsl:template match="DETAILS">
  <tr><th colspan="2" align="left">&#160;<i><xsl:value-of
select="name()" /></i></th></tr>
  <tr><xsl:apply-templates select="RATING"/></tr>
  <tr><xsl:apply-templates select="VIDEOSIZE"/></tr>
  <tr><xsl:apply-templates select="ENCODERATE"/></tr>
  <tr><xsl:apply-templates select="LENGTH"/></tr>
</xsl:template>

<xsl:template match="HISTORY">
  <tr><th colspan="3" align="left">&#160;<i><xsl:value-of
select="name()" /></i></th></tr>
  <tr><xsl:apply-templates select="WORKFLOW"/></tr>
  <tr><xsl:apply-templates select="NOTES"/></tr>
</xsl:template>

<xsl:template match="FILESIZE">
  <td>&#160;&#160;&#160;&#160;<small><xsl:value-of
select="name()" /></small></td>
  <td><xsl:value-of select="self::FILESIZE"/></td>
  <td rowspan="11" valign="top" align="right">
    <xsl:element name="IMG">
      <xsl:attribute name="width">160</xsl:attribute>
      <xsl:attribute name="border">0</xsl:attribute>
      <xsl:attribute name="src"><xsl:value-of
select="document('global/settings.xml')//settin
gs/thumbnail_url/text()" />/image_<xsl:value-of
select="//ATTRIBUTES/HEADER/CONTENTID"/></xsl:v

```

```

        alue-of
        select="//ATTRIBUTES/HEADER/THUMBNAIL"/></xsl:a
        ttribute>
        <xsl:attribute name="alt">Thumbnail is
        unavailable</xsl:attribute>
    </xsl:element>
</td>
</xsl:template>

<xsl:template match="COPYRIGHT">
    <td>#160;#160;#160;#160;<small><xsl:value-of
    select="name()" />:</small></td>
    <td><xsl:value-of select="self::COPYRIGHT"/></td>
</xsl:template>

<xsl:template match="AUTHOR">
    <td>#160;#160;#160;#160;<small><xsl:value-of
    select="name()" />:</small></td>
    <td><xsl:value-of select="self::AUTHOR"/></td>
</xsl:template>

<xsl:template match="DESCRIPTION">
    <td>#160;#160;#160;#160;<small><xsl:value-of
    select="name()" />:</small></td>
    <td><xsl:value-of select="self::DESCRIPTION"/></td>
</xsl:template>

<xsl:template match="OWNER">
    <td>#160;#160;#160;#160;<small><xsl:value-of
    select="name()" />:</small></td>
    <td><xsl:value-of select="self::OWNER"/></td>
</xsl:template>

<xsl:template match="CREATEDT">
    <td>#160;#160;#160;#160;<small><xsl:value-of
    select="name()" />:</small></td>
    <td><xsl:value-of select="self::CREATEDT"/></td>
</xsl:template>

<xsl:template match="RATING">
    <td>#160;#160;#160;#160;<small><xsl:value-of
    select="name()" />:</small></td>
    <td><xsl:value-of select="self::RATING"/></td>
</xsl:template>

<xsl:template match="VIDEOSIZE">
    <td>#160;#160;#160;#160;<small><xsl:value-of
    select="name()" />:</small></td>
    <td><xsl:value-of select="self::VIDEOSIZE"/></td>
</xsl:template>

<xsl:template match="ENCODERATE">
    <td>#160;#160;#160;#160;<small><xsl:value-of
    select="name()" />:</small></td>
    <td><xsl:value-of select="self::ENCODERATE"/></td>
</xsl:template>

```



```

<xsl:template match="LENGTH">
  <td>#160;#160;#160;#160;<small><xsl:value-of
    select="name()" />:</small></td>
  <td><xsl:value-of select="self::LENGTH"/></td>
</xsl:template>

<xsl:template match="WORKFLOW">
  <td colspan="3">#160;#160;<xsl:value-of select="name()" />:<br/>
    <xsl:for-each select="WORK">
      #160;#160;#160;#160;<small><xsl:value-of
        select="@time"/>#160;#160;#160;#160;<xsl:value-of
        select="@user"/>:<small>#160;#160;#160;#160;<xsl:value-
          of select="self::WORK"/></small><BR/>
      </xsl:for-each>
    </td>
</xsl:template>

<xsl:template match="NOTES">
  <td colspan="3">#160;#160;<xsl:value-of select="name()" />:<br/>
    <xsl:for-each select="NOTE">
      #160;#160;#160;#160;<small><xsl:value-of
        select="@time"/>#160;#160;#160;#160;<xsl:value-of
        select="@user"/>:<small>#160;#160;#160;#160;<xsl:value-
          of select="self::NOTE"/></small><BR/>
      </xsl:for-each>
    </td>
</xsl:template>

</xsl:stylesheet>

```

**FIGURE 4**

```

<?xml version="1.0" encoding="UTF-8" ?>

- <!--
THIS IS A TEMPLATE XML FILE FOR ASSETS ATTRIBUTES
-->
- <ATTRIBUTES>
- <HEADER>
- <STATUS>1</STATUS>
  <FILESIZE>12384</FILESIZE>
  <DESCRIPTION />
  <AUTHOR>Igor</AUTHOR>
  <OWNER>Igor</OWNER>
  <COPYRIGHT>Sequoia Broadband, Inc.</COPYRIGHT>
  <CREATEDT>2002-08-30</CREATEDT>
  <THUMBNAIL>Ocean_View.jpg</THUMBNAIL>
</HEADER>
- <DETAILS>
- <LENGTH>00:10:00</LENGTH>
  <ENCODERATE />
  <VIDEOSIZE />
  <RATING />
</DETAILS>
- <HISTORY>
- <WORKFLOW>
- <WORK user="cn=igor,o=collisionNetwork,c=us"
  time="8/13/02 10:41 AM">Asset is created from System
  Admin GUI.</WORK>
  <WORK user="cn=igor,o=collisionNetwork,c=us"
    time="8/20/02 2:58 PM">Performed Mark Error</WORK>
  <WORK user="cn=igor,o=collisionNetwork,c=us"
    time="8/20/02 2:59 PM">Performed Clear Error</WORK>
</WORKFLOW>
  <NOTES />
</HISTORY>
</ATTRIBUTES>

```

FIGURE 5

Home

Manage Assets

Manage Playlists

Manage Jobs

Manage Remote Sites

Manage NOC

Help

Create

View

Workflow

Create Asset

Add Metadata: Ocean.mpg

HEADER:

FILESIZE

DESCRIPTION

AUTHOR

OWNER

COPYRIGHT

CREATEDT

THUMBNAIL

DETAILS:

LENGTH

ENCODERATE

VIDEOSIZE

RATING

HISTORY:

FIGURE 6

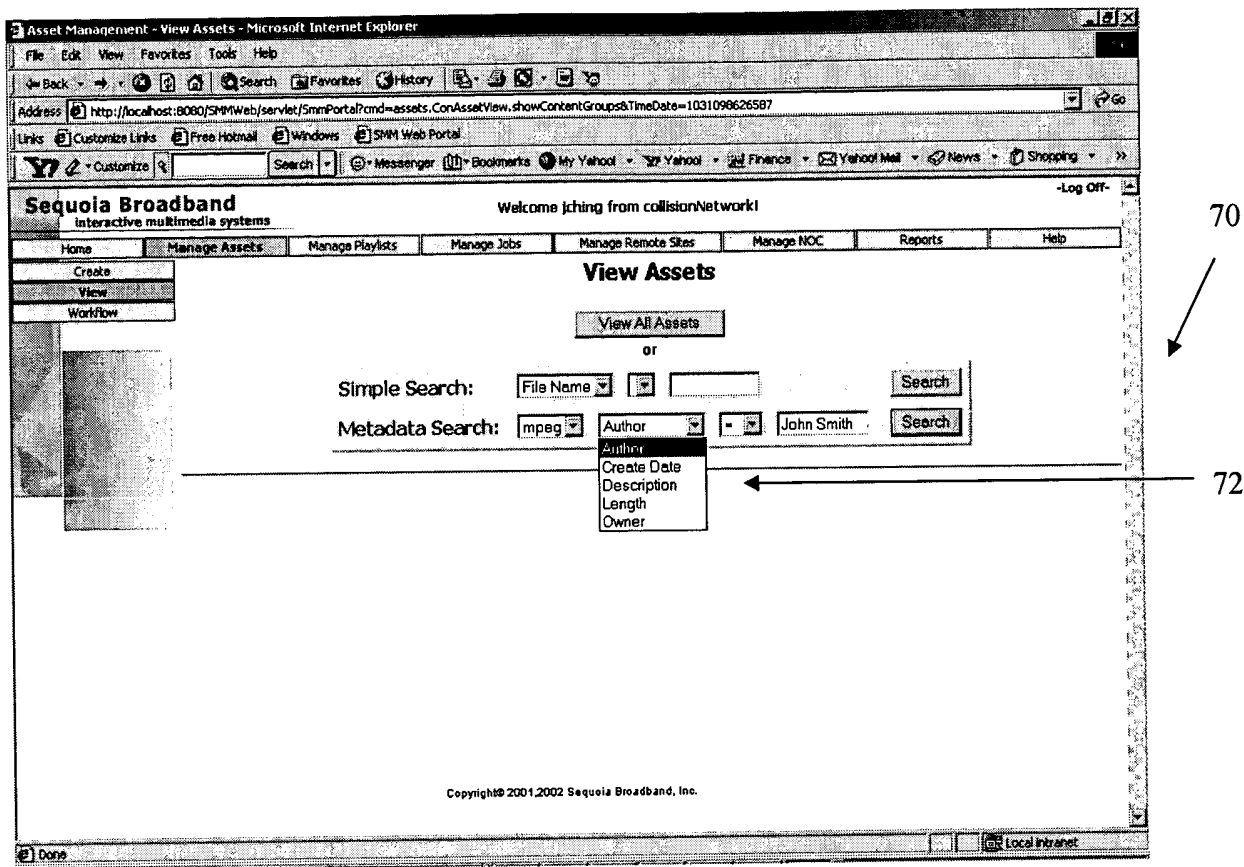
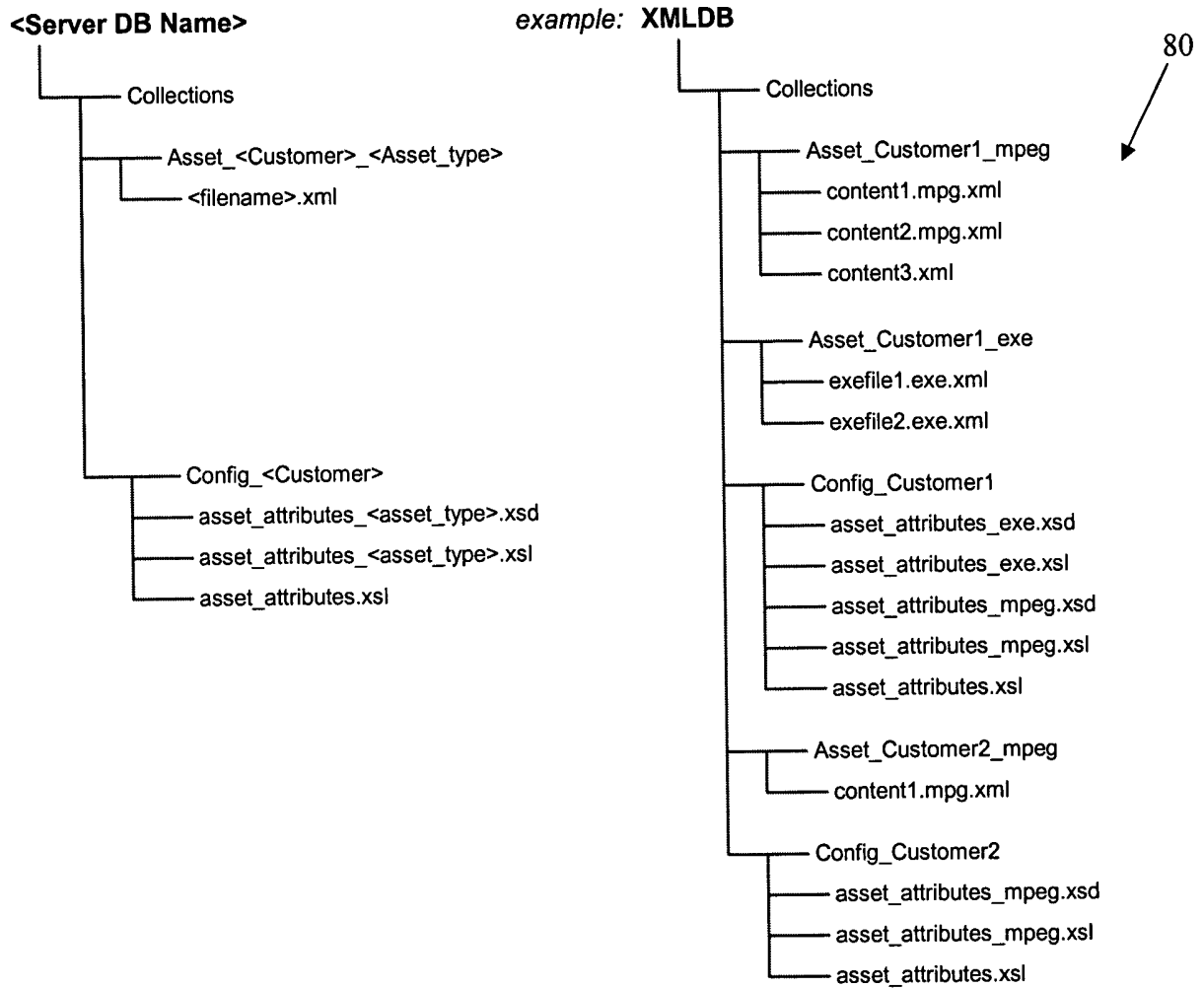


FIGURE 7

## XML Database Collection Structure



**FIGURE 8**

# Asset Creation Process Flow

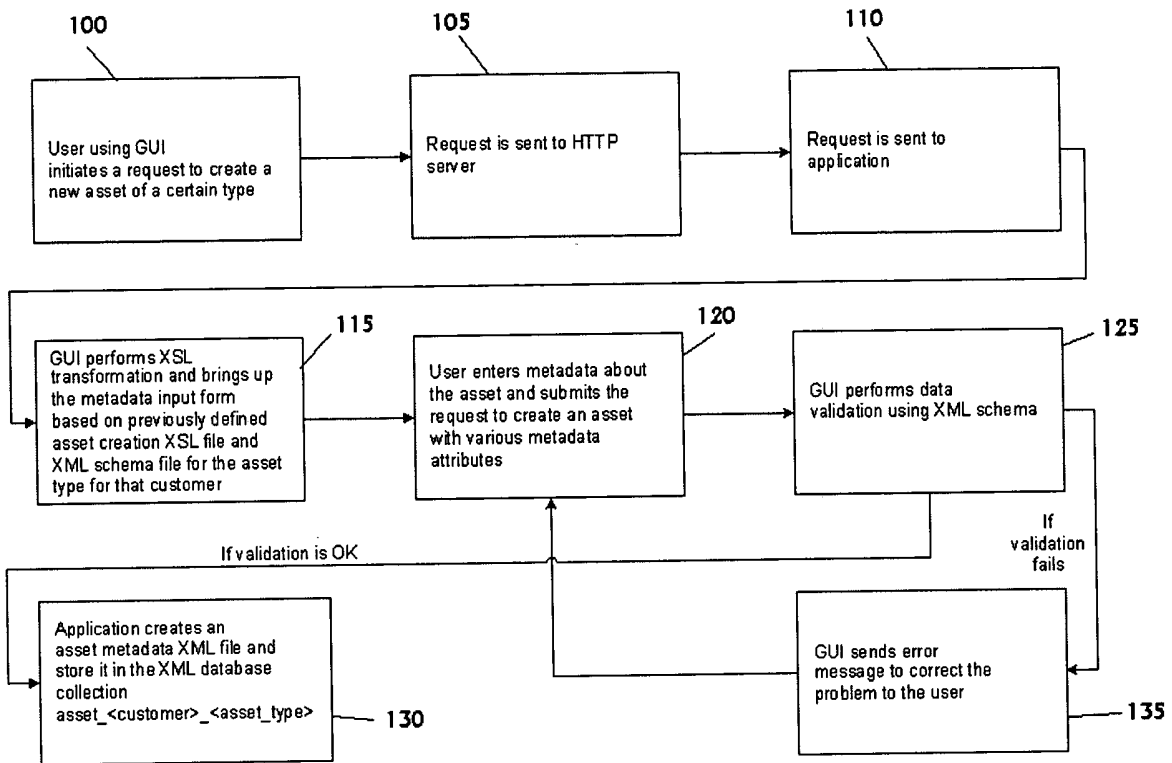
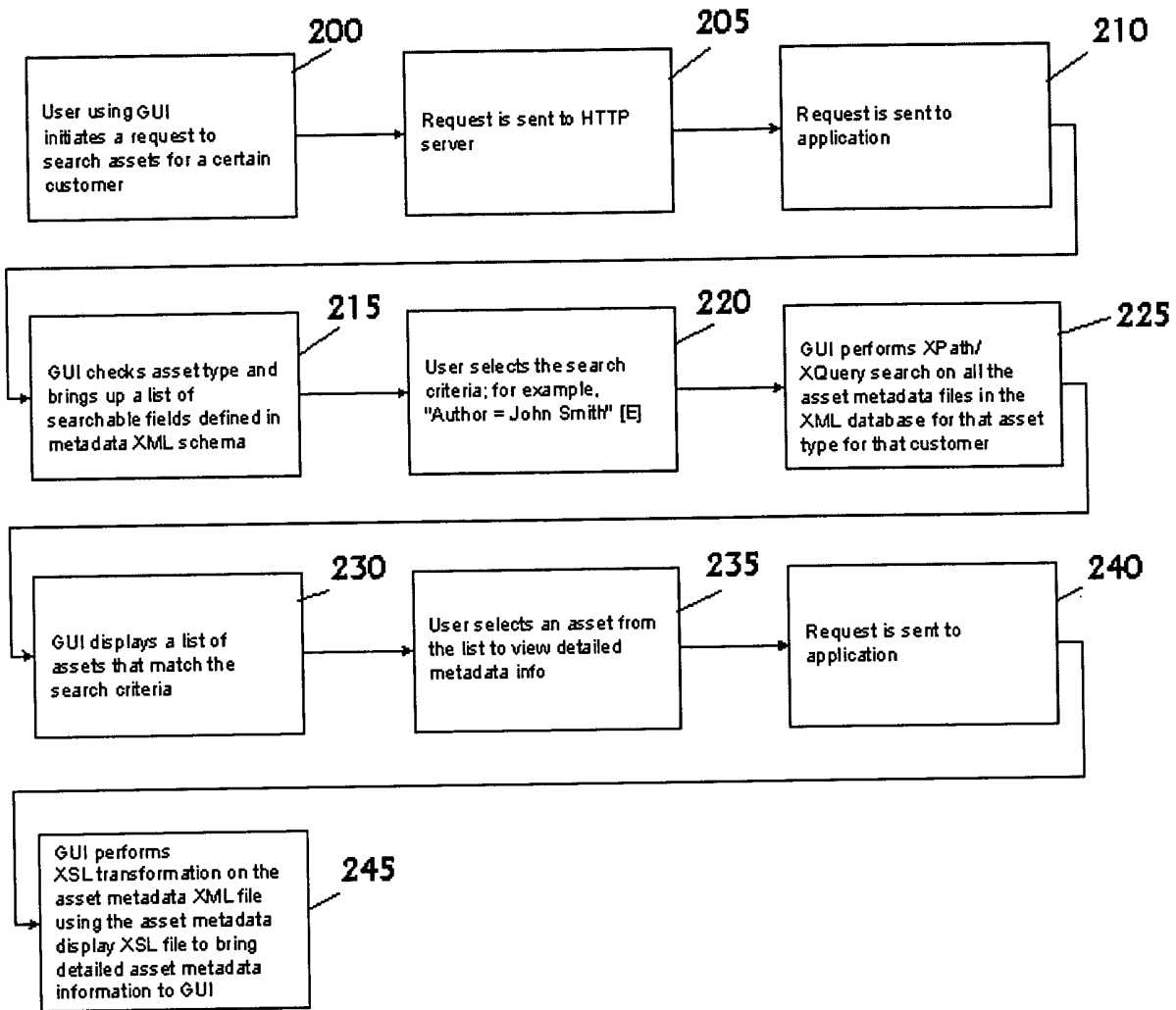


FIGURE 9

### Asset Search Process Flow



**FIGURE 10**

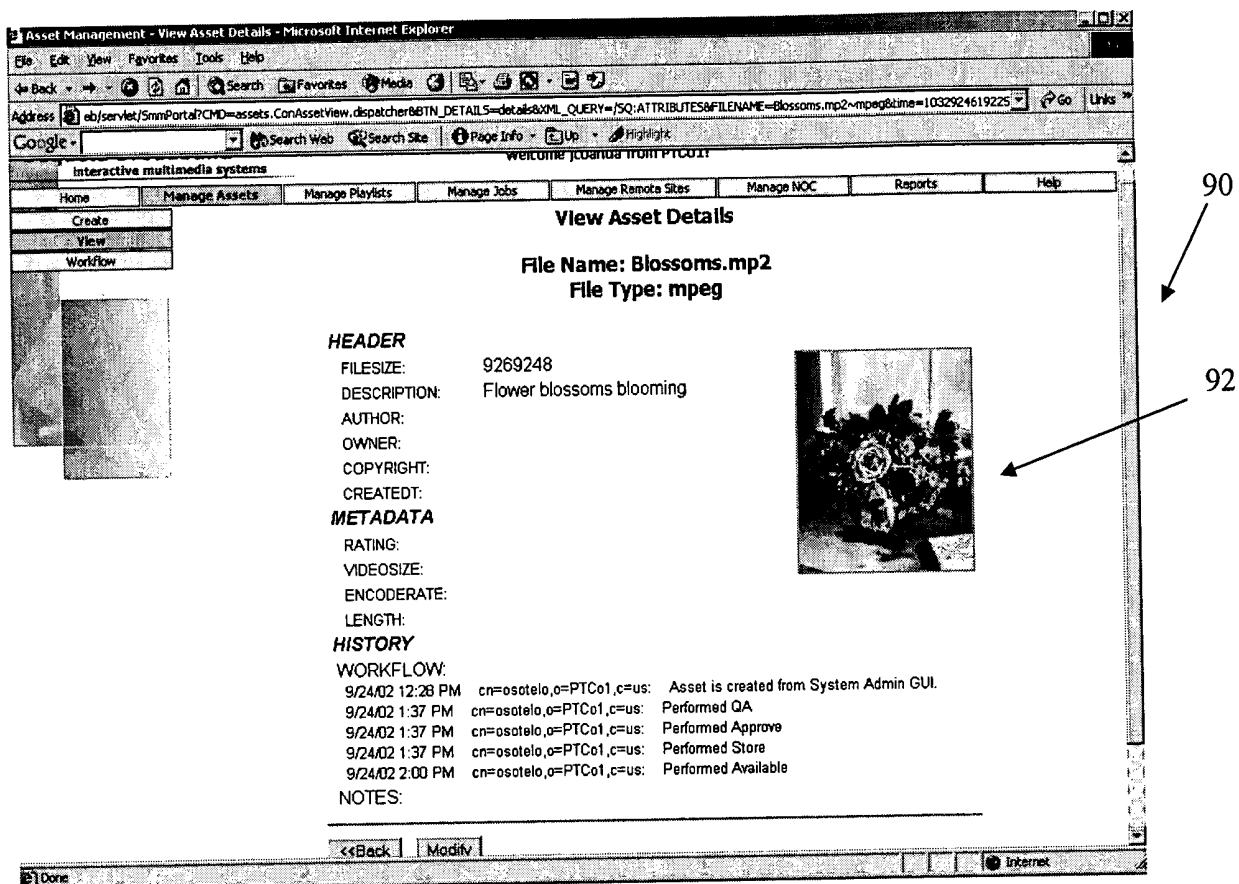


FIGURE 11